Fine-grained Visual Dribbling Style Analysis for Soccer Videos with Augmented Dribble Energy Image

Runze Li Bir Bhanu
Center for Research in Intelligent Systems
University of California, Riverside, Riverside, CA - 92521
rli047@ucr.edu, bhanu@cris.ucr.edu

Abstract

Recent advances in interpretations of soccer are predominantly made through analyzing high-level contents of soccer videos. This work targets on these highlight actions and movements in soccer games and it focuses on dribbling skills performed by the top players. Our work leverages understanding of complex dribbling video clips by representing a video sequence with a single Dribble Energy Image(DEI) that is informative for dribbling styles recognition. To overcome the shortage of labelled data, this paper introduces a dataset of soccer video clips from Youtube, employs Mask-RCNN to segment out dribbling players and OpenPose to obtain joints information of dribbling players. Besides, to solve issues caused by camera motions in highlight soccer videos, our work proposes to register a video sequence to generate a single image representation DEI and dribbling styles classification. Our approach can achieve an accuracy of 87.65% on dribbling styles classification and it is observed that data augmentation using joints-reasoned GAN can improve the classification performance.

1. Introduction

Computer vision has been employed in sports analysis for broadcasting usage and commercial application. Techniques connected with players identification, actions recognition and score prediction are critical scenarios in baseball, soccer, ice hockey, etc. In Europe, five top football leagues, Premier League, La Liga, Championnat de France de football Ligue 1, Bundesliga and Lega Serie A, organize highest-level soccer games every year and attract soccer players around the world to participate. Among those, Premier League is the most profitable league, which achieves a revenue of 5297 million pounds in 2016-2017 [25]. The booming business value in soccer drives deeper analysis targeting on players, coaches, tactics, etc, to obtain precise and elaborate statistics of every soccer player and match.

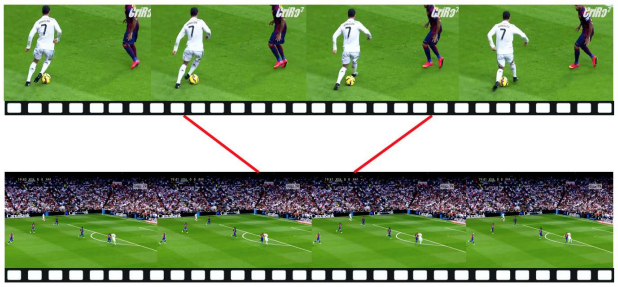

Figure 1. Top: highlights video recorded by a close camera which captures detailed motion of soccer players. Bottom: overview video recorded by remote cameras. The goal of this work is how to determine the dribbling style in a sequence of frames from a soccer game.

Investigating dribbling skills in soccer games, is beneficial to both clubs to train their players, and for defenders to know how to improve defending skills. For example, top players, like Cristiano Ronaldo, Lionel Messi, Neymar Jr., are well-known as icons for their smart dribbling skills, which help them evade through defenders and score in soccer games. Soccer fans are constantly amazed by adept dribbling skills and are curious to understand and analyze "What dribbling styles are the players showing?" when they watch top-class soccer games like World Cup, UEFA European Champions, etc.

Typically, soccer games are recorded with cameras distributed around the soccer field resulting players movements far from a camera view and being recorded at a small scale. These kinds of videos can be utilized by coaches and players to study tactics of different soccer teams according to the positions of players. While, there are also highlight videos captured in close-view to players where the camera is moving to track players who are controlling the ball and performing various tasks that may require significant skills. These videos illustrate expert skills in soccer games and they are shared by fans around the world. However, in soccer video analysis, the main challenge is the lack of labeled

data in both overview videos and highlight videos. Even if highlight videos are captured, another issue is the camera calibration in tracking and capturing players who are performing smart movements. Players move at a super-fast speed and cameras also are trying to catch up with players, causing camera motions. A sequence of frames with both temporal and spatial information will be messed up and can provide little information without image registration. The fact is that massive data available to us does not have corresponding camera parameters for calibration. To solve above problems, we collect highlights of video clips from Youtube which involve different dribbling skills performed by professional soccer players in both real games, and simulated game environment, like FIFA and PEPS games on Xbox, PlayStation and tutorial video clips provided by soccer fans. Deep neural network models trained on large-scale image dataset are used to localize and segment soccer players in every frame in each video clip. We propose an affine-transformation-based approach to register a sequence of frames with target dribbling players into a single image representation called as Dribble Energy Image (DEI). Fianlly, Convolutional Neural Network is trained to classify dribbling styles and conditional GAN with constraints on body posture is employed for data augmentation. Main contributions of this paper are :

- Collect and build up a soccer dribbling dataset involving data with variants from multiple sources.
- Introduce Dribble Energy Image (DEI) to transfer a sequence of frames to an image representation using affine-transformation-based image registration method which can handle raw video clips at multi-scale resolution and solve camera motion problems.
- Classification of soccer dribbling styles using Convolutional Neural Network and train Generative Adversarial Network to augment dataset for improving the classification performance.
- Construct dribbling player's joints model as probability conditions for training Conditional GAN to generate DEI where objects are guided to follow the embedding of a soccer player's body.

2. Related Work

Computer vision already plays a key role in sport analysis ranging from basketball, soccer, baseball and ice hockey based on large amounts of streaming data. They produce statistics of events in a game by either analyzing videos captured by cameras or captured semantic data.

Research is increasingly focused on soccer video analysis including video summarization, event classification and action recognition. Efros *et al.* [6] recognized actions at a distance in soccer matches by introducing a motion descriptor based on optical flow in a spatio-temporal volume for each human figure. Baccouche *et al.* [1] proposed an approach for 4-classes action classification in soccer videos using a recurrent neural network. Tsunoda *et al.* [24] proposed a hierarchical-LSTMs to conduct action recognition involving "Dribble", "Shoot" and "Pass" actions in fustal and their dataset was collected by 14 calibrated and synchronized cameras distributed in a futsal field. Cioppa *et al.* [5] proposed a bottom-up approach to interpret soccer games captured by the main camera stream. Their method extracted features from soccer videos and corresponding features with semantic meanings for better events understanding in soccer games. Jiang *et al.* [12] employed the CNN for feature extractions and combined RNN to emphasize temporal information to detect events in soccer videos. Theagarajan *et al.* [22] conducted soccer analysis for identifying players who has the ball in soccer matches by using Convolutional Neural Network and GAN for data augmentation. Our work is directed for exploring details in depth in soccer videos, specifically, with insights on how to recognize dribbling styles of a player who is controlling the ball and performing smart dribbling actions. As far as we know, we are the first one concentrating on the work of fine-grained dribbling styles classification and our work can handle video clips with different resolutions.

To perform soccer video analysis, tracking, recognizing and identifying players are the initial steps. Soccer players can be detected and extracted using object detection techniques. Girshick [7] proposed the Fast R-CNN for detecting object (persons) and this work was optimized to the Mask R-CNN [10]. However, occlusions on players makes it a challenging task to detect players in soccer games and dribbling actions are invisible due to occlusions. After detecting the soccer players in each frame, we extract the 2D pose information of each player using OpenPose proposed by Cao *et al.* [4].

3. Framework

In this section, we describe the framework of our approach for processing and augmenting data. Figure 2 shows the overview architecture of our approach, which consists of dribbling player's segmentation, pose detection, body parts, image registration, data augmentation and dribbling styles classification modules.

3.1. Dribbling Player Segmentation

In our framework, we employ Mask R-CNN [10] to localize and segment players who are performing dribbling skills from each frame in every video sequence. The Mask R-CNN extends the Faster R-CNN and adopts a two-stage procedure, which predicts not only the class label and bounding box of an object, but also a binary mask of each ROI in parallel. The model we use is pre-trained on the Microsoft COCO dataset [16] involving classes like persons,

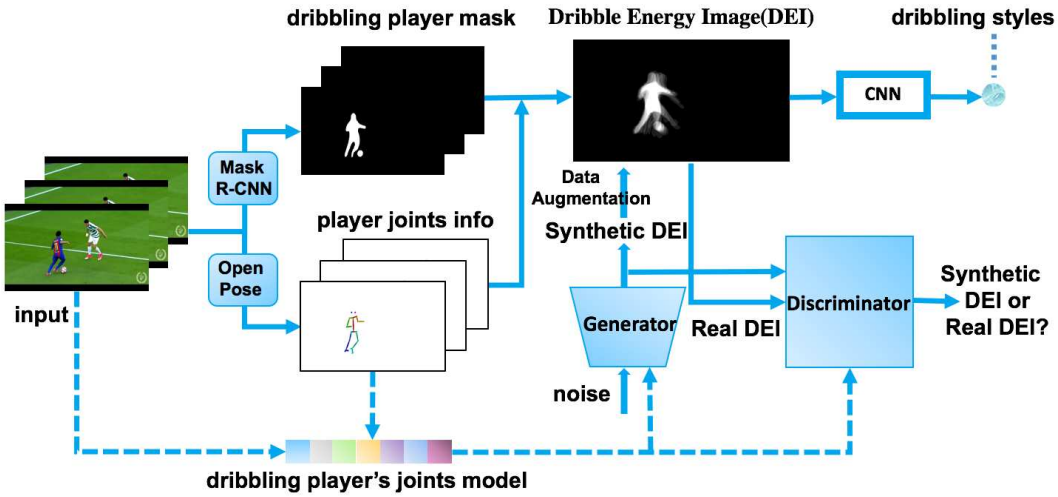

Figure 2. Architecture for the Classification of Soccer Dribbling Styles. Video clips are collected from YouTube. Each video sequence is processed to generate dribbling players' mask and pose. DEI representation is generated after image registration and it is used for the classification of dribbling styles with data augmentation from GAN.

sports ball, which are dominant targets in soccer. Our approach processes every frame of each video clip through Mask R-CNN. In each frame, we only keep the masks of the player who is dribbling and the soccer ball. The processing time of Mask R-CNN is **3.79 seconds / frame** using one NVIDIA GPU at a resolution of 480 $\times$ 854. Images of higher resolutions require more processing time. The bottom left part in Figure 3 shows the mask results of the dribbling player on a video sequence using Mask R-CNN.

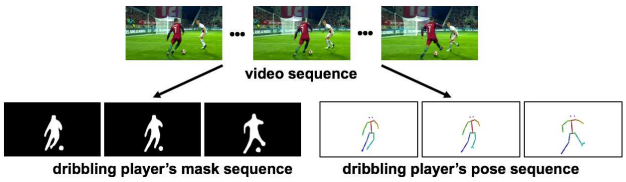

Figure 3. Top: video sequence of dribbling actions performed by Cristiano Ronaldo. Bottom left: mask of dribbling player in the video sequence. Bottom right: pose of dribbling player in the video sequence.

3.2. Dribbling Player Pose Detection

After localizing and segmenting visual soccer players in the video sequence, we use OpenPose [4] to extract 2D pose information of target soccer players. OpenPose takes a color image of size $w \times h$ as the input and produces the 2D locations of anatomical key-points for each person in the image as the output. In this paper, we use pose described with 25 keypoints pre-trained on Microsoft COCO[16] dataset. For every frame of each video clip, we use OpenPose to obtain key-points of soccer players with mask of soccer players obtained from Mask R-CNN and we only use coordinates of joints of the only one player who is performing the dribbling action. The processing time of Openpose is **0.1825 seconds / frame** using one NVIDIA GPU with the resolution of 720 $\times$ 1080. The bottom right part of Figure 3 illustrates detected pose of the dribbling player in a video sequence using OpenPose.

3.3. Dribbling Player Image Registration

In this section, we describe the approach for image registration. As dribbling actions are performed in a sequence of frames by soccer players, it causes spatial motions of objects within each frame and camera motions across consecutive images. Most of current work processes a sequence of frames in a spatial stream, a temporal stream and combine the two streams which requires massive computations in both time and memory. Bilen *et al.* [2] introduced a dynamic image which is a standard RGB image that summarizes the appearance and dynamics of a whole video sequence so that it can be used for action recognition. Han *et al.* [9] proposed Gait Energy Image, as a spatial temporal gait representation for human walking recognition. Bobick *et al.* [3] proposed both binary motion-energy image (MEI) to represent where motion has occurred in an image sequence and motion-history image (MHI) which is a scalar-valued image where intensity is a function of motion. Our work is similar to GEI. Compared with MEI and MHI, our approach implements frame registration to solve the motion caused by the dribbling player and the camera. The dribble energy image encodes the spatial-temporal information of a video sequence into a single image which enables CNN to be trained and tested in a faster and easier way. To eliminate influences imposed by camera motions, we propose the hip-

joint-based and the affine-transformation-based registration methods to transform a sequence of frames into the same embedding so as to generate a single image representation for each dribbling video clip.

3.3.1. Hip-joint-based image registration

When watching soccer players moving and performing dribbling actions, we observe that the hip area around the player's body is relatively static in reference to whole body and lower torso mainly supports movements of the body. Based on this observation, we make the assumption that image registration across a sequence of frames can be done by taking the hip joint of soccer player as the reference. Therefore, we embed the mask image sequence to generate one energy image according to the coordinate of the hip joint of dribbling player in the video sequence. The right side of upper branch in Figure 4 illustrates the result using hip-joint-based image registration method.

3.3.2. Affine-transformation-based image registration

We refine the image registration process by proposing the affine-transformation-based method for DEI for better registration results and the process is illustrated in Figure 5. As described above, we have a sequence of mask images $I_j, j \in 1, 2, 3, ..., n$ containing the dribbling player from every video clip $V_i, i \in 1, 2, 3, ..., N$. Meanwhile, in each mask image, the pose information with 25 key-points is bounded with the target dribbling player. Then we take the first mask image I_1 as the base mask image and highlight left hip (LH_1) and right hip (RH_1) coordinates of the dribbling player, described as (x^1_{lb}, y^1_{lb}) and (x^1_{rb}, y^1_{rb}) along (x, y) axes, respectively. We also name each image $I_j, j \in 2, 3, 4, ..., n$ as the sequence mask image starting from the second mask image. To conduct image registration, we take the second sequence mask image I_2 as an example, firstly, we localize left hip (LH_2) and right hip (RH_2) coordinates of the target dribbling player, described as (x^2_{ls}, y^2_{ls}) and (x^2_{rs}, y^2_{rs}) respectively. We would like to align the sequence mask image I_2 according to the middle points (Mid_1 and Mid_2 in Figure 5) in the line connected by $[LH_1,\ RH_1]$ and $[LH_2,\ RH_2]$, which are $[(x^1_{lb}, y^1_{lb}), (x^1_{rb}, y^1_{rb})]$ and $[(x^2_{ls}, y^2_{ls}), (x^2_{rs}, y^2_{rs})]$ in Figure 5, respectively. Therefore, in each sequence mask image $I_j, j \in 2, 3, ..., n$ we would align the mid point $Mid_j, j \in 2, 3, ..., n$ to $Mid^{'}_j, j \in 2, 3, ..., n$ using the transformation matrix T_A described by equation 1. Then we calculate distance between aligned middle point $Mid^{'}_j, j \in 2, 3, ..., n$ and middle point in base mask image Mid_1 and register each sequence mask image on the base mask image to generate DEI.

$$Mid^{'}_j = T_A \times Mid_j, j \in 2, 3, 4, ..., n \tag{1}$$

We calculate affine transformation for the transformation marix T_A using every pair of the sequence mask image and the base mask image. Considering how we process the sequence mask image I_2 as an example, we find a point $(\hat{x}^1, \hat{y}^1)$ in base mask image I_1 to construct the equilateral triangle among three points, (x^1_{lb}, y^1_{lb}), (x^1_{rb}, y^1_{rb}) and $(\hat{x}^1, \hat{y}^1)$. Following the same way, in the sequence mask image I_2, we find a point $(\hat{x}^2, \hat{y}^2)$ to construct a equilateral triangle among three points, (x^2_{lb}, y^2_{lb}), (x^2_{rb}, y^2_{rb}) and $(\hat{x}^2, \hat{y}^2)$. Then we calculate the affine transformation using these two sets of three-points reference equation 2. In 2, (x_1, y_1) is the end point from equilateral triangle in the base mask image and (x_2, y_2) is the end point from equilateral triangle in the sequence mask image.

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix} \times \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} \tag{2}$$

The affine-transformation-based registration method is shown in Figure 5. The DEI of affine-transformation-based method is illustrated in right side of top branch in Figure 4. The intermediate registration results of applying affine-transformation-based method gradually across a video sequence are presented in bottom in Figure 4.

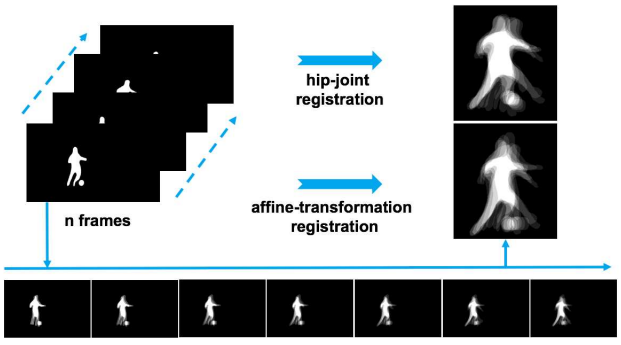

Figure 4. Top: Image registration results for DEI generated by two methods. Bottom: intermediate registration results by applying affine-transformation-based method gradually on a video sequence.

3.4. Dribbling Styles Classification via Convolutional Neural Network

To classify dribbling styles using DEIs, we use Convolutional Neural Network(CNN) [14] to perform training and testing. We experiment our approach using AlexNet [15], VGG-16 [21] and ResNet18 [11]. We compare these networks and find the network that can achieve the best classification accuracy and generalizability. We split dataset consisting of DEIs into training and testing datasets respectively. We train the model with training dataset in which each image is resized to 224×224. We choose a mini-batch

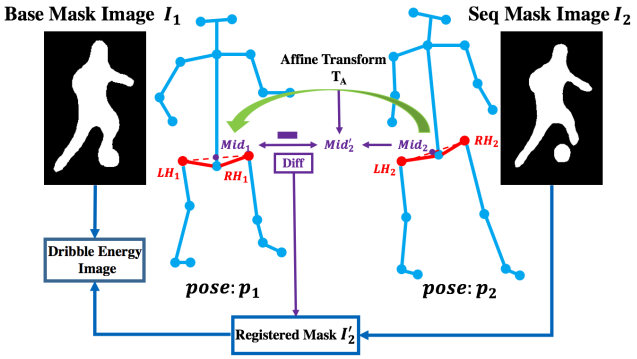

Figure 5. Affine-transformation-based image registration for DEI. Left: key-points of the dribbling player bounded by base mask image. Right: key-points of the dribbling player bounded with sequence mask image. T_A is calculated using key-points and DEI is generated by aligning sequence mask image on base mask image.

size to train and during every epoch the training data is randomly shuffled. we terminate the training session when we observe the loss converges to avoid over-fitting problems.

3.5. Dataset Augmentation via Generative Adversarial Network

Generative Adversarial Network(GAN) proposed by Goodfellow *et al.* [8] has shown great advantages in image generation, translation, animation, etc. This work was improved by Radford *et al.* [19] by employing convolutional layers, batch normalization layers of deep neural network in both generator and discriminator to create a novel architecture, called DCGAN. Mirza *et al.* [17] proposed the Conditional Generative Adversarial Network(cGAN) by giving a conditional vector along with the random noise to the generator and to the discriminator together with an image.

To solve the shortage of soccer videos with dribbling actions, we employ DCGAN for data augmentation to our training dataset. The purpose of data augmentation is to determine whether generating more variability to the training dataset can help to improve the performance of our framework.

3.5.1. Dataset Augmentation via DCGAN

We train a Deep Convolutional Generative Adversarial Network (DCGAN) [19] for data augmentation. DCGAN contains two deep convolutional neural networks, one generator and one discriminator. The generator will accept a random noise vector z and output an image by learning the mapping of data space as $I_z = G(z; \theta_g)$. The discriminator accepts a real or a generated image alternately and outputs a probability of which sources the input image is from. The discriminator is trained to maximize the probability $logD(\mathbf{x})$ of identifying correct labels to both training images and images generated from generator. Also, the generator is trained simultaneously to minimize $log(1-D(G(z)))$. We follow suggestions provided by [19] by replacing pooling layers with convolutional layers in both generator and discriminator and using batch normalization and Leaky ReLU as the activation function. Then we design and train the DCGAN to generate DEIs for usage in our training dataset.

3.5.2. Dribbling player joints model reasoned GAN

To reason how different poses of soccer players perform during dribbling actions in soccer games, we build the dribbling player's joints model based on joints information we obtain from OpenPose [4]. As illustrated in Figure 6, each body segment, $\mathbf{i}$, is approximated by a 2D limb with parameter $\mathbf{l^i}$: the limb length. The main body is defined via joints, neck and mid-hip, which are used to calculate the global position. The articulated structure of the dribbling player body has a total of 41 degrees of freedom (DOFs) with two descriptions: global position and local position. The global position is described by the angle of the torso formed by the neck and the mid-hip within the Cartesian coordinate system, as $\mathbf{g} = \theta^0$. The local position is calculated for each limb length and joints angle as $\mathbf{l}$ and $\boldsymbol{\theta}$:

$$\mathbf{l} = [l^i], i = 1, ..., 20 \tag{3}$$

$$\boldsymbol{\theta} = [\theta^j], j = 1, ..., 20 \tag{4}$$

For local position, as shown in Figure 6, we calculate 20 limb lengths and normalize them. We calculate 20 angles between each pair adjacent joints. For example, in Figure 6 (a), angle $\theta_{j\sim i}$ is calculated for limb l_j and l_i. We concatenate each angle after the limb length to form a 41 DOFs vector $[\mathbf{l}\ \mathbf{g}\ \boldsymbol{\theta}]^\intercal$ to describe dribbling pose of soccer players.

For each dribbling styles, we calculate the mean of

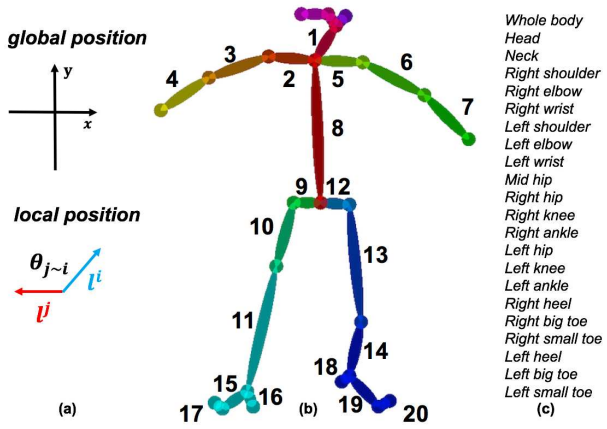

Figure 6. Soccer dribbling Player Joints Model. (a): global position in coordinate system and local position between adjacent joints. (b) & (c): limb segments in soccer players.

players' joints model as the condition vector. We give the conditional vector to both generator and discriminator and design the Conditional-GAN to generate DEIs of different dribbling styles. We expect dribbling players joints model would guide and formalize the result from generator to be within the embedding of the soccer player. Therefore, the dribbling players joints model works as the prior condition for the generator to learn data mapping, and loss function will also help the generator to refine data mapping it has learned. This architecture is shown in Figure 7.

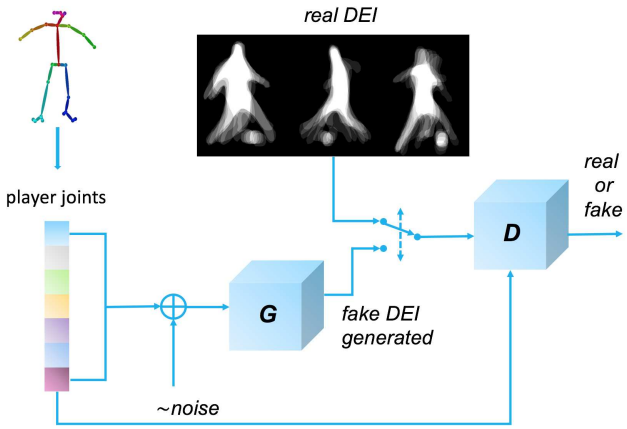

Figure 7. Dribbling Player Joints Model is used with Conditional-GAN.

4. Experiments Results

We train and evaluate our approach with dataset we have collected. We implement our framework using PyTorch [18] on a workstation with 4 NVIDIA 1080-Ti GPUs.

4.1. Dataset

We collect and setup our dataset by searching and crawling on Youtube. We concentrate on highlights of soccer videos and cut a long highlight video into pieces with dribbling actions. After downloading videos from Youtube, we clip each video into tiny video clips and each video clip contains one dribbling style. In total, our dataset consists of 313 video clips with more than 6400 frames and each each video clip is annotated with the corresponding dribbling style name: STEPOVER, ELASTICO and CHOP. Dribbling styles annotations are terminologies used in soccer games. The ***STEPOVER*** is the style where soccer players will use their non-dominant foot to pretend kicking the ball to one direction but go over the ball in actual to evade defenders. The ***ELASTICO*** is the style where soccer players use outside of their dominant foot to push the ball to one direction, then change to move to reverse direction with ball. The ***CHOP*** is the style where soccer players use one foot to kick the ball to the reverse direction behind their body. Table 1 shows basic statistics of our dataset and Figure 8 shows examples in our dataset. Our dataset is setup with following features:

1. Dribbling players in our dataset are from almost all top clubs from 5 top leagues in Europe and there are more than 55 players identities with dribbling actions, including female players.
2. Dataset contains data from synthetic games, FIFA on Xbox One and PlayStation which are vivid as real data.
3. Data is in high resolution, 143 video clips are in 1080 $\times$ 1920 and 96 clips are in 720 $\times$ 1920.

Dribbling Styles	STEPOVER	ELASTICO	CHOP
Total clips	123	81	110
Total frames	1434	2301	2697
Average # of frames	11.6	28.4	24.5
Teams	14	11+	25+
Number of Players	16	21	50+

Table 1. Soccer Dribbling Dataset Statistics

Figure 8. Examples from our dataset. First two rows: STEPOVER style. Middle two rows: ELASTICO style. Botton two rows: CHOP style.

4.2. Dribbling Styles Classification

In this section, we present dribbling styles classification results obtained using our framework, and compare classification results on several main-stream architectures using DEIs generated from two methods separately. For each video clip, we process each frame with Mask R-CNN and OpenPose. Then we use two image registration methods to generate the DEI on each video sequence. We use DEIs as representations of video clips in which soccer players are performing dribbling actions to perform dribbling styles classification with Convolutional Neural Networks. We use 216 video clips which are initially collected for training and 59 video clips for testing with 5-fold cross validation mechanism. From Table 2, we observe that using transformation-based DEIs on ResNet-18 achieves the best performance, this method servers as our baseline.

Method	Mean Accuracy	STDEV.
Hip*_AlexNet[15]	85.24%	0.31%
Affine*_AlexNet[15]	87.8%	**2.2%**
Affine_VGG-15[21]	83.73%	4.08%
Affine_ResNet-18[10]	**88.14%**	3.17%

Table 2. Soccer Dribbling Styles Classification Results. *Hip and *Affine: registration methods.

4.3. Data Augmentation

This section, describes how we implement and observe how augmenting dataset affects the performance of our task by training the DCGAN model and players-joints-model-reasoned Conditional-GAN model, respectively.

To train the DCGAN, the generator was designed to accept a 1×100 noise vector which is randomly sampled from a normal distribution. The output of generator is a grayscale image of size 64×64. The discriminator accepts a grayscale image of size 64×64 from either real images or generated images as input and predicts whether the image is real or generated by generator. We train the DCGAN with a learning rate of 2×10^{-4} and a mini-batch size of 18. We optimize both generator and discriminator using the Adam optimization and Binary Cross Entropy loss function [19]. Generated DEIs are shown in the top row in Figure 9.

For conditional GAN reasoned by dribbling player's joints model, we concatenate the dribbling players joints model as the vector mentioned in Section 3.5.1 to random noise as the input to the generator. The random noise is generated in 1×100 dimensions from a normal distribution. The output of generator is a grayscale image of size 64×64. To train the discriminator, we give a grayscale image from either real images or generated images alternatively of size 64×64 as the input. We concatenate the dribbling player joint model as the vector with the last layer of discriminator and let discriminator predict whether the input image is the real or the generated. We train our model with similar mechanism as described in [19]. Generated DEIs are shown in the bottom row in Figure 9. From Figure 9, we observe that using DCGAN can generate good DEIs for different dribbling styles, but there is a lot of noise in the background. From the results of GAN guided by dribbling players joints model, we observe that the contrast of the player against background is much higher and there is much lower noise in the background.

We augment the training dataset using generated data and observe how it affects the classification performance. We take the DEIs obtained from transformation-based registration method as the ***base dataset***. We train the ResNet-18 following the same experiment set up and evaluate the performance on the testing dataset. We compare the performance on base dataset, base dataset augmented with DCGAN and Conditional GAN by dribbling player's joints model reasoned, respectively. We use 5-fold cross validation mechanism to explore the sensitivity of data augmentation. Results are shown in Table 3.

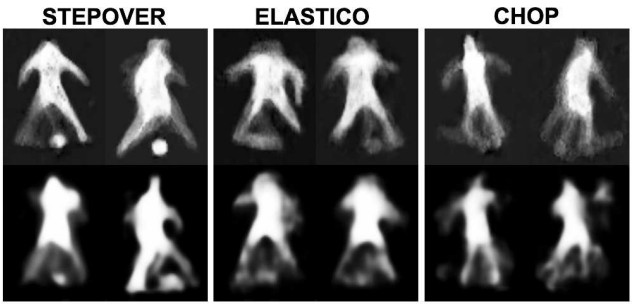

Figure 9. DEIs generated by two types of GAN. Top row: results from DCGAN. Bottom row: results from dribbling players joints model used with conditional-GAN. Column: both images in each column are for the same dribbling style.

Dataset	Mean Accuracy	STDEV.
Base	88.14%	2.78%
Base+DCGAN_30	88.47%	3.26%
Base+JointsGAN_30	**89.83%**	**1.7%**
Base+JointsGAN_60	88.47%	2.21%

Table 3. Soccer Dribbling Styles Classification Results with Data Augmentation. Base+DCGAN_30 means augmenting base dataset with 30 generated data from DCGAN.

From Table 3, we can see that by augmenting 30 DEIs of each dribbling styles generated by DCGAN to training dataset, the mean accuracy of dribbling styles classification can be improved to **88.47%** with **3.26%** in standard deviation of 5-fold cross validation. Adding the same amount of DEIs generated by Conditional GAN guided by dribbling player's joints model, the classification accuracy is **89.83%** with **1.69%** in standard deviation. Comparing the results without data augmentation, we observe that augmenting dataset for training can help improve the accuracy of respective networks. Besides, we observe that using dribbling player's joints model as the condition to the GAN can decrease the standard deviation (from **3.26%** to **1.69%**) of 5-fold cross validation. This observation aligns with the assumption that providing the dribbling player's joints model to the GAN can help the GAN generate data within the embedding of soccer players. However, when we add 60 generated DEIs in each dribbling styles, no improvements are observed. We analyze that the reason is that samples generated using GAN have low variants and maintain not all details as real samples. So even more sampled are used, these samples are still very similar to each other.

4.4. Comparisons on Video Classification Methods

In this section, we compare our framework with other video classification methods. The first method we compare is the one described in [13]. We extract fixed number of frames from each video sequence and stack these frames as the early fusion as the input to the CNN, which we call video-level 2D-CNN framework. The second method we compare is to use 3D-CNN network [23] which takes the fixed number of frames as the input for video classification, which we call 3D-CNN framework. We use 4 frames and 6 frames extracted from each video clip respectively in our experiments. The third method that we compare with is the two-stream network [20], which explores spatial information from one RGB image in a sequence and utilizes temporal information calculated by optical flows from every two consecutive frames in a sequence. Furthermore, we extract the player who is performing dribbling and form a tracklet from each dribbling video clip. We use these tracklet sequences as the input to both the two stream network and 3D-CNN again for dribbling styles classification. The last experiment we do is we combine features from tracklets using a stream of 3D-CNN and features from DEIs using a stream of CNN. Then we use combined features as the input to the last layer in CNN for classification. We split the dataset into **4** folds and test with 4-fold cross validation mechanism. After split, there are 90 video clips of "STEPOVER", 60 of "ELASTICO" and 81 of "CHOP" in each split of the training data. The rest of 33 video clips of "STEPOVER", 21 of "ELASTICO" and 29 of "CHOP" in each split are used as the testing data.

In Table 4, we report classification performance using different approaches with 4-fold cross validation mechanism. "Affine" means DEIs are generated using transfomration-based method for registration. From Table 4, we can see that, via employing DEIs as input to the deep network, our approach achieves classification accuracy of **87.65%** on the average and **2.78%** as the standard deviation of dribbling styles classification by using ResNet-18 network. Another observation is that by using features extracted from DEIs combined with features from tracklets via 3D-CNN can promote classification accuracy from **64.33%** using single 3D-CNN with input of dirbbling player tracklets to **85.97%**, which illustrates that DEIs indeed contain vital information to capture dribbling styles. The reasons for the poor performance of 3D CNN and two stream CNN are: (1) A single image in spatial stream or even part of a video sequence can not represent a dribbling style or even a dribbling action so that features extracted from parts of raw RGB images can be used to inference dribbling styles; (2) 3D-CNN and temporal stream is hard to train to generalize features to represent dribbling styles. Because without dribbling players registration, the motion of the dribbling player across the video sequence and the motion of each part of human body within each frame are quite different, which causes feature-points-based transformation and optical flow to be inaccurate. Therefore, using our framework, all frames in the video sequence are registered and utilized to represent a complete dribbling action, and registration based on observation that hip area of the dribbling player is static ensures the motion represented by DEI across a video sequence is accurate.

Method	Mean Accuracy	STDEV.
Video_2D_CNN [13]	54.21%	4.29%
Video_3D_CNN(4*) [23]	53.61%	3.49%
Video_3D_CNN(6*) [23]	52.11%	15.16%
Spatial_Stream [20]	59.15%	4.51%
Temporal_Stream [20]	61.58%	6.17%
Spatial_Tracklet [20]	57.93%	6.18%
Temporal_Tracklet [20]	57.92%	3.78%
3D*_CNN_Tracklet [23]	64.33%	**2.08%**
DEI+3D_CNN_Tracklet	**85.97%**	2.54%
DEI_ResNet18	**87.65%**	2.78%

Table 4. Soccer Dribbling Styles Classification Results. *4 and *6: number of frames sampled from a video sequence as the input to the 3D-CNN

5. Conclusions

This paper uses DEI and CNNs to classify dribbling styles of soccer players. The DEI is a single image containing spatio-temporal information of a dribbling video sequence. We perform image registration to eliminate the camera motion. Generative models are used to augment dataset during the training session. To formalize the generative model to generate data within the specific embedding in our dataset, this paper proposes the soccer dribbling player's joint model to guide the generative networks. The results show that our approach achieves an accuracy of 87.65% in fine-grained dribbling styles classification. With the help of dribbling player's joints model as the condition to the GAN, the accuracy of classification is improved from 88.14% to 89.83%. Experiments on 3D-CNN taking the dribbling player's tracklet as the input, after using features extracted from DEIs to perform late fusion, the classification accuracy is improved from 64.33% to 85.97%. Future work will include extending current dataset to involve more dribbling styles and other sports, and enabling dribbling player detection automatically taking occlusions, viewpoint variants into consideration.

6. Acknowledgments

This work was supported in parts by Bourns Endowment and funds and a gift from SEVAai to the University of California at Riverside.

References

[1] Moez Baccouche, Franck Mamalet, Christian Wolf, Christophe Garcia, and Atilla Baskurt. Action classification in soccer videos with long short-term memory recurrent neural networks. In *Proceedings of the 20th International Conference on Artificial Neural Networks: Part II*, ICANN'10, pages 154–159, Berlin, Heidelberg, 2010. Springer-Verlag.

[2] H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, and S. Gould. Dynamic image networks for action recognition. In *CVPR*, 2016.

[3] A. F. Bobick and J. W. Davis. The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):257–267, March 2001.

[4] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017.

[5] Anthony Cioppa, Adrien Deliege, and Marc Van Droogenbroeck. A bottom-up approach based on semantics for the interpretation of the main camera stream in soccer games. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

[6] Alexei A. Efros, Alexander C. Berg, Greg Mori, and Jitendra Malik. Recognizing action at a distance. In *IEEE International Conference on Computer Vision*, pages 726–733, Nice, France, 2003.

[7] Ross Girshick. Fast R-CNN. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2015.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[9] Ju Han and Bir Bhanu. Individual recognition using gait energy image. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(2):316–322, Feb. 2006.

[10] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[12] H. Jiang, Y. Lu, and J. Xue. Automatic soccer video event detection based on a deep neural network combined cnn and rnn. In *2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 490–494, Nov 2016.

[13] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[15] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[17] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.

[18] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

[19] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2015.

[20] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 568–576. Curran Associates, Inc., 2014.

[21] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[22] Rajkumar Theagarajan, Federico Pala, Xiu Zhang, and Bir Bhanu. Soccer: Who has the ball? generating visual analytics and player statistics. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

[23] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[24] Takamasa Tsunoda, Yasuhiro Komori, Masakazu Matsugu, and Tatsuya Harada. Football action recognition using hierarchical lstm. *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 155–163, 2017.

[25] Bill Wilson. Premier league club revenues soar to 4.5bn.